Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computing system, comprising:

a first approximation apparatus to approximate ~~the~~ a term 2^X, wherein X is a real number, the first approximation apparatus comprises a rounding apparatus to accept an input value (X) that is a real number represented in floating-point format, and to compute a rounded value ($\lfloor X \rfloor_{integer}$) by rounding the input value (X) toward minus infinity, wherein the rounded value ($\lfloor X \rfloor_{integer}$) is represented in an integer format;

a memory to store a computer program that utilizes the first approximation apparatus; and

a central processing unit (CPU) to execute the computer program, the CPU is cooperatively connected to the first approximation apparatus and the memory.

a1

2. (Cancelled).

3. (Currently Amended) The system of claim 1, wherein the first approximation apparatus includes:

an integer-to-floating-point converter to accept as input a first rounded value ($\lfloor X \rfloor_{integer}$), being an input value (X) that is a real number represented in an integer format, and to convert the first rounded value ($\lfloor X \rfloor_{integer}$) to a second rounded value ($\lfloor X \rfloor_{floating\text{-}point}$) represented in floating-point format.

4. (Original) The system of claim 1, wherein the first approximation apparatus includes:

a floating-point subtraction operator to compute the difference between an input value (X) and $\lfloor X \rfloor_{floating\text{-}point}$ which is the input value (X) rounded toward minus infinity and is represented in floating-point format.

5. (Currently Amended) The system of claim 1, wherein the first approximation apparatus includes a shift-left logical operator to generate a shifted $\lfloor X \rfloor_{integer}$ value by shifting a rounded value ($\lfloor X \rfloor_{integer}$), being an input value (X) that is a real number to the left by a predetermined number of bit positions.

6. (Original) The system of claim 1, wherein the first approximation apparatus includes:

a second approximation apparatus to accept ΔX as input, to approximate $2^{\Delta X}$, and to return an approximation of $2^{\Delta X}$, wherein $\Delta X = X - \lfloor X \rfloor_{floating\text{-}point}$ and $\lfloor X \rfloor_{floating\text{-}point}$ is the input value (X) rounded toward minus infinity and is represented in floating-point format.

7. (Original) The system of claim 6, wherein the second approximation apparatus computes the approximation of $2^{\Delta X}$ by applying Horner's method in calculating a sum of a plurality of elements of a series in the equation $2^{\Delta X} = \sum_{N=0}^{\infty} \frac{(\Delta X \ln 2)^N}{N!}$.

a1

8. (Currently Amended) The system of claim 1, wherein the first approximation apparatus includes:

an integer addition operator to accept a shifted $\lfloor X \rfloor_{integer}$ value, being an input value (X) that is a real number represented in an integer format and undergoes a bit-wise shift operation and an approximation of $2^{\Delta X}$ as input, and to perform an integer addition operation on the shifted $\lfloor X \rfloor_{integer}$ value and the approximation of $2^{\Delta X}$ to generate an approximation of 2^X, wherein $\Delta X = X - \lfloor X \rfloor_{floating\text{-}point}$ and $\lfloor X \rfloor_{floating\text{-}point}$ is the input value (X) rounded toward minus infinity and is represented in floating-point format.

9. (Original) The system of claim 1, further comprising:

a third approximation apparatus to approximate a term C^Z, wherein C is a constant and a positive number and Z is a real number,

the third approximation apparatus using a floating-point multiplication operator to compute a product of $\log_2 C \times Z$, and feeding the product of $\log_2 C \times Z$ into the first approximation apparatus to generate an approximation of C^Z.

10. (Original) A method comprising:

generating a first rounded value and a second rounded value;

subtracting the second rounded value from an input value (X) to generate ΔX;

generating an approximation of $2^{\Delta X}$;

performing a bit-wise left shift to the first rounded value to generate a shifted value; and

approximating 2^X by performing an integer addition operation to add the shifted value to the approximation of $2^{\Delta X}$.

11. (Original) The method of claim 10, wherein generating the first rounded value comprises:

rounding an input value (X) downward to generate the first rounded value represented in an integer format.

a1

12. (Original) The method of claim 10, wherein generating the second rounded value comprises:

converting the first rounded value represented in an integer format to the second rounded value represented in floating-point format.

13. (Original) The method of claim 10, wherein generating an approximation of $2^{\Delta X}$ comprises:

applying Horner's method in calculating a sum of a plurality of elements of a series in the equation $2^{\Delta X} = \sum_{N=0}^{\infty} \frac{(\Delta X \ln 2)^N}{N!}$.

14. (Original) The method of claim 10, wherein performing a bit-wise left shift operation to the first rounded value comprises:

shifting the first rounded value to the left by a predetermined number of bit positions so that the first rounded value occupies bit positions reserved for an exponent of a floating-point value.

15. (Original) The method of claim 10, wherein approximating 2^X comprises:

performing an integer addition operation to add the shifted value to the approximation of $2^{\Delta X}$, such that the first rounded value is added to an exponent value of the approximation of $2^{\Delta X}$.

16. (Original) A machine-readable medium comprising instructions which, when executed by a machine, cause the machine to perform operations comprising:

a first code segment to perform computations to approximate the term 2^X, wherein X is a real number.

17. (Original) The machine-readable medium of claim 16, wherein the first approximation apparatus includes:

a second code segment to accept an input value (X) that is a real number represented in floating-point format, to compute a rounded value ($\lfloor X \rfloor_{integer}$) by rounding the input value (X) toward minus infinity, and to return the rounded value ($\lfloor X \rfloor_{integer}$) which is represented in an integer format.

a1

18. (Original) The machine-readable medium of claim 17, wherein the second code segment computes the approximation of $2^{\Delta X}$ by applying Horner's method in calculating a sum of a plurality of elements of a series in the following equation, $2^{\Delta X} = \sum_{N=0}^{\infty} \frac{(\Delta X \ln 2)^{N}}{N!}$.

19. (Original) The machine-readable medium of claim 16, wherein the first code segment includes:

a third code segment to accept ΔX as input and to generate an approximation of $2^{\Delta X}$, wherein $\Delta X = X - \lfloor X \rfloor_{floating\text{-}point}$ and $\lfloor X \rfloor_{floating\text{-}point}$ is the input value (X) rounded and is represented in floating-point format.

20. (Currently Amended) The machine-readable medium of claim 16, wherein the first code segment includes:

a fourth code segment to accept a shifted $\lfloor X \rfloor_{integer}$ value, being an input value (X) that is a real number represented in an integer format and undergoes a bit-wise shift operation and an approximation of $2^{\Delta X}$ as input, and to generate an approximation 2^X by performing an integer addition operation on the shifted $\lfloor X \rfloor_{integer}$ value and the approximation of $2^{\Delta X}$, wherein $\Delta X = X - \lfloor X \rfloor_{floating\text{-}point}$ and $\lfloor X \rfloor_{floating\text{-}point}$ is the input value (X) rounded and is represented in floating-point format.

a1

21. (Original) The machine-readable medium of claim 16, further includes:

a fifth code segment to approximate a term C^Z, wherein C is a constant and a positive number and Z is a real number, the fifth code segment computing a product of $\log_2 C \times Z$ and feeding the product of $\log_2 C \times Z$ into the first code segment to generate an approximation of C^Z.